Towards a Software Prototype Supporting Automatic Recognition of Sketched Business Process Models

Manuel Zapp, Peter Fettke and Peter Loos

Institute for Information Systems at the German Research Center for Artificial Intelligence (DFKI GmbH) and Saarland University, Saarbrücken, Germany
{manuel.zapp,peter.fettke,peter.loos}@iwi.dfki.de

Abstract. The paper at hand presents a prototype that implements an approach to recognize the structure of business process models sketched on paper and whiteboards to create digital versions of those models. We explain the different steps from common photos of business process models sketched on paper or whiteboards to their digital version. We modified existing approaches from sketch recognition to fulfill the needs of sketched business process model recognition. Therefore, a dataset was generated that in turn is used to train different classifiers for different shapes within business process models.

Keywords: sketch recognition, BPM, computer vision, machine learning.

1 Introduction

In Business Process Modeling there has always been an interest in building tools assisting the modeler in creating business process models. Such tools aim at being user-friendly and intuitive. Nevertheless, pen and paper is still the predominant choice, especially in the early design phase of business processes. With no doubt using pen and paper is one of the most intuitive ways of creating business process models. Without manual work, these models can only be stored as pictures or archived on paper and not be transferred into existing modelling systems. To bridge this gap, we introduce a software system that fully automatically transfers sketched business processes into a digital format that can be interpreted by common modeling tools. This prototype is developed within the research project INDIGO in cooperation with a leading Business Process Management software provider, which has multiple customers that have expressed a strong interest in the functionalities of the prototype. Although we use the modeling convention Event-driven Process Chain (EPC) as a sample, our approach is generic and can be applied to other modelling conventions.

In our work, we want to implement an offline approach that is capable of constructing digital business process models from sketched ones on whiteboards or paper. Please note that this is work in progress. In this paper we concentrate on the structural recognition of business process models without solving the Hand Writing Recognition problem for labels that are usually present in sketched process models.

13th International Conference on Wirtschaftsinformatik,
February 12-15, 2017, St. Gallen, Switzerland

Zapp, M.; Fettke, P.; Loos, P. (2017): Towards a Software Prototype Supporting Automatic Recognition of Sketched Business Process Models, in Leimeister, J.M.; Brenner, W. (Hrsg.): Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik (WI 2017), St. Gallen, S. 1283-1286

2 Related Work

In general, two approaches in sketch recognition can be distinguished – offline approaches and online approaches. In online approaches the process of sketching is observed, whereby the temporal order information of strokes is leveraged to reduce the search space [1]. Offline approaches are applied to images of finished sketches only and thus are much more challenging.

A lot of work has been done in the area of online sketch recognition, especially in the field of flowchart recognition [2–4], that is by means related to business process recognition. There is only one work available in that area that deals with an offline approach [1] and we refer to it.

To the best of our knowledge we are pioneers on the field of sketched business process digitalization. There exist some works that use post-it®s and create their own modeling conventions that are designed and simplified for later digitalization.[1] The major drawbacks of these systems are the modeling conventions that first have to be learnt by all persons involved as well as the limitations of the modeling conventions and the additional hardware that is needed in form of post-it®s.

3 Our Approach and Its Implementation

3.1 Overview

The prototype is designed as a web application that provides interfaces to prepare data in form of photos taken from sketched business process models to train the software system itself. Additionally, an interface is provided that allows a user to upload images of sketched process models and responds with a representation of the recognized model. Components that actually incorporate image processing and computer vision functionalities are built in C++, due to performance reasons and better handling of the OpenCV[2] library that is used. In the following the necessary steps for an automated recognition are explained.

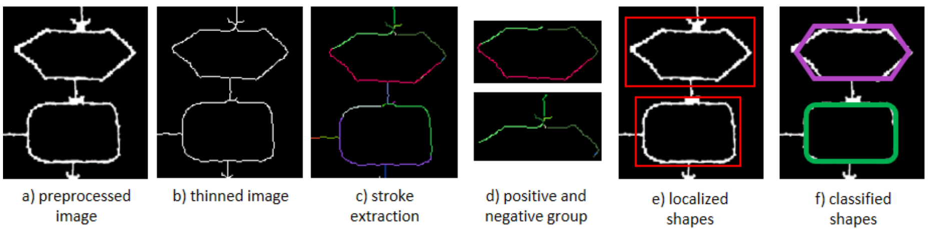

Figure 1. Different steps of processing

[1] http://www.symbioworld.com/de/produkt/produkt-portfolio/workshop-methode.html
[2] http://opencv.org/

3.2 Preprocessing and Stroke Extraction

In our work we target an end-to-end scenario. As we do not want to have unrealistic expectations at the initial quality of the images that should be processed, we allow the images to come from common hardware like smartphone cameras, but we demand sufficiently high contrast between the color of the used pen and the color of the surface the sketch is drawn on. As a consequence, some preprocessing to the initial image is necessary. We apply common filters from Computer Vision to get rid of artifacts in the image. The result of this process is a clean image where only white lines are displayed on a black background as shown in Figure 1a).

Since we follow an offline approach there is no structural information about strokes available. Based on the approach presented in [5] we reconstruct strokes from pixel data and adapt it to our needs. The authors propose to apply a Morphological Thinning algorithm [6] to the preprocessed image, so that all lines are reduced to a one-pixel width as shown in Figure 1b). The extracted strokes are sets of single pixels from the thinned image. Since these strokes may span over several different shapes, which is undesirable for our work we introduce a splitting algorithm. We trace the extracted strokes and at every *n-th* point and perform a principal component analysis on the last and the next *k* pixels of the actual stroke. The principal component analysis gives us the orientation of the last *k* and the next *k* pixels. If the angle between them exceeds an experimentally learnt threshold of 70 degrees, the stroke is split at that point. So all strokes extracted belong solely to one single shape in the process model as shown in Figure 1c).

3.3 Localization

To actually classify shapes into the right category we need to know which strokes form a shape. Common approaches extract rectangular regions from the image and test each against a classifier. Usually shapes contain text that leads to different appearances within the same type of shape. Thus we follow a technique presented in [1] that is called shapeness estimation. The authors propose a feature to construct a fast classifier that solely decides whether a set of strokes forms a shape or not. A group of strokes is a set of strokes that fulfils predefined constraints like spatial ones. The authors build the groups by breadth-first-search. We adopt their work for classical flow-chart recognition to also be able to detect sketched EPCs. Therefore, a dataset of 108 sketched EPCs was created by 36 people with expertise in business process modeling, all sketching 3 different EPCs including 24 shapes on average. The shapes were then extracted so we end up with nearly 2.600 labelled samples. Figure 1d) shows two stroke sets, where one forms a valid shape and, thus, got accepted by the shapeness classifier and an invalid and, thus, rejected one. In Figure 1e) the localized shapes are displayed.

3.4 Classification

The classification problem can be solved in different ways. We decided to choose the Histogram of Oriented Gradients feature since it is commonly used in sketch recognition [7]. We obtained best results in combination with Support Vector Machines. We train a classifier for each of the seven shapes (i.e. events, functions, application systems, organizational units and the three connector types) that are applied in the respective order to proposed stroke sets. If the maximum confidence exceeds a learnt threshold, the shape of the classifier with highest confidence is chosen otherwise the stroke set is dropped. Table 1 shows the accuracy measures for the overall model. We used 10-fold cross-validation to validate these results.

Table 1. Accuracy Measures

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
0.969	0.017	0.938	0.965	0.951	0.930

4 Conclusion and Discussion

We presented a prototype that implements an approach to construct common business process models based on the EPC modeling notation from sketched process models in terms of their structure. Our prototype is research in progress. At the moment, there is just a first evaluation of our prototype available. These first evaluation activities show that our approach is promising. Further evaluation is planned.

References

1. Wu, J., Wang, C., Zhang, L., Rui, Y.: Offline sketch parsing via shapeness estimation. In: IJCAI International Joint Conference on Artificial Intelligence (2015).
2. Bresler, M., Phan, T. Van, Prusa, D., Nakagawa, M., Hlavac, V.: Recognition System for On-Line Sketched Diagrams. Proc. Int. Conf. Front. Handwrit. Recognition, ICFHR. 2014–Decem, 563–568 (2014).
3. Lemaitre, A., Mouchère, H., Camillerapp, J., Coüasnon, B.: Interest of syntactic knowledge for on-line flowchart recognition. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 7423 LNCS, 89–98 (2013).
4. Carton, C., Lemaitre, A., Couasnon, B.: Fusion of statistical and structural information for flowchart recognition. Proc. Int. Conf. Doc. Anal. Recognition, ICDAR. 1210–1214 (2013).
5. Rajan, P., Hammond, T.: From Paper to Machine: Extracting Strokes from Images for use in Sketch Recognition. (2008).
6. Lü, H.E., Wang, P.S.P.: A comment on "a fast parallel algorithm for thinning digital patterns." Commun. ACM. 29, 239–242 (1986).
7. Hu, R., Collomosse, J.: A Performance Evaluation of Gradient Field HOG Descriptor for Sketch Based Image Retrieval. (2013).